Codigo configuración

/\* USER CODE BEGIN PV \*/

uint32\_t value\_adc = 0;

uint32\_t value\_dac = 0;

uint16\_t ValPrint[5];

/\* USER CODE END PV \*/

/\* USER CODE BEGIN 2 \*/

HAL\_DAC\_Start(&hdac, DAC\_CHANNEL\_1);

HAL\_DAC\_SetValue(&hdac, DAC\_CHANNEL\_1, DAC\_ALIGN\_12B\_R, value\_dac);

HAL\_ADC\_Start\_IT(&hadc1);

/\* USER CODE END 2 \*/

/\* USER CODE END WHILE \*/

HAL\_GPIO\_TogglePin(LD2\_GPIO\_Port, LD2\_Pin);

**itoa** (value\_adc,ValPrint,10);

HAL\_UART\_Transmit(&huart3, ValPrint, 5, 0xFFFF);

HAL\_UART\_Transmit(&huart3, "\n", 1, 0xFFFF);

HAL\_Delay(100);

HAL\_ADC\_Start\_IT(&hadc1);

/\* USER CODE BEGIN 3 \*/

/\* USER CODE BEGIN 4 \*/

**void** **HAL\_ADC\_ConvCpltCallback**(ADC\_HandleTypeDef \*hadc) {

value\_adc = HAL\_ADC\_GetValue(hadc);

HAL\_DAC\_SetValue(&hdac, DAC\_CHANNEL\_1, DAC\_ALIGN\_12B\_R, value\_dac);

value\_dac = value\_dac + 10;

**if**(value\_dac>4095) {

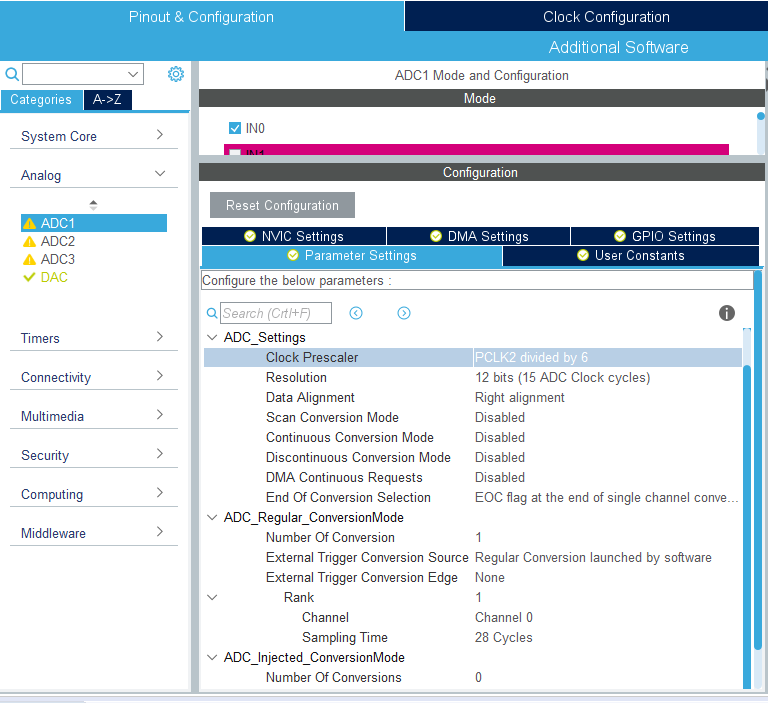
value\_dac = 0;

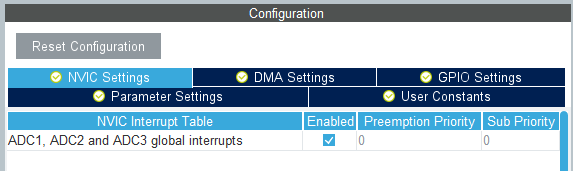
}

}

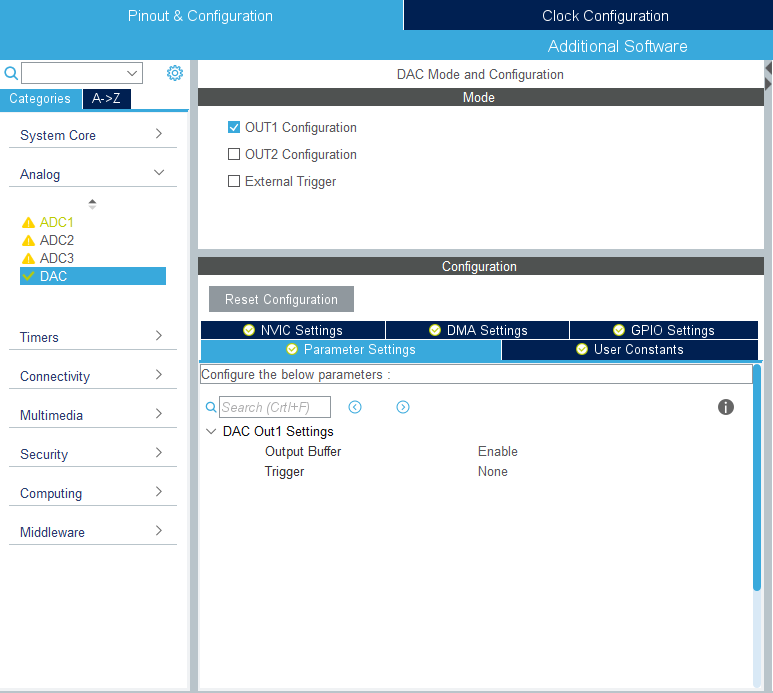
/\* USER CODE END 4 \*/

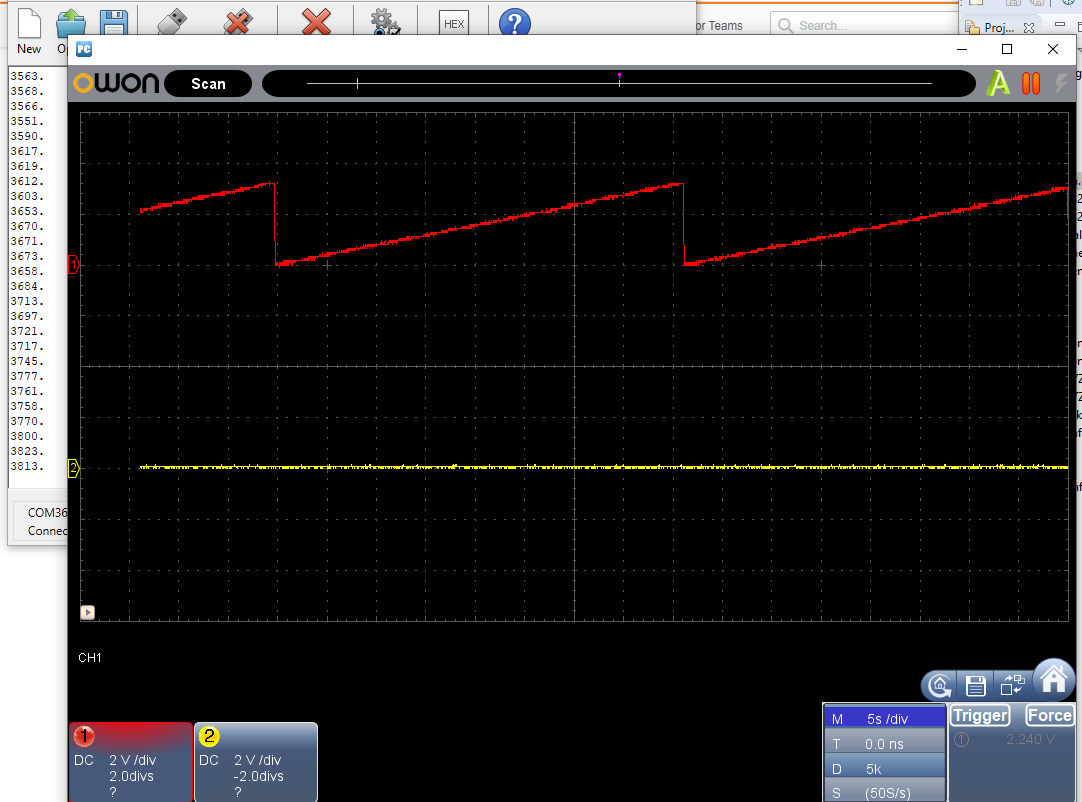
Configuración ADC





Configuración DAC





DAC

/\* USER CODE BEGIN Includes \*/

**#include** "math.h"

/\* USER CODE END Includes \*/

/\* USER CODE BEGIN PV \*/

uint32\_t valDAC = 0;

**float** value = 0;

uint32\_t t = 0;

/\* USER CODE END PV \*/

/\* USER CODE BEGIN 2 \*/

HAL\_DAC\_Start(&hdac, DAC\_CHANNEL\_1);

HAL\_DAC\_SetValue(&hdac, DAC\_CHANNEL\_1, DAC\_ALIGN\_12B\_R, valDAC);

/\* USER CODE END 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

**while** (1)

{

/\* USER CODE END WHILE \*/

value = 2000\***sin**(2\*3.14159\*t\*0.001) + 2000;

valDAC = (uint32\_t) value;

HAL\_DAC\_SetValue(&hdac, DAC\_CHANNEL\_1, DAC\_ALIGN\_12B\_R, valDAC);

t++;

HAL\_Delay(1);

/\* USER CODE BEGIN 3 \*/

}

